Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

OPTIMIZING COMPLEX INTERACTION DYNAMICS IN CRITICAL INFRASTRUCTURE WITH A STOCHASTIC KINETIC MODEL

Fan Yang
Alina Vereshchaka
Wen Dong

Department of Computer Science and Engineering
State University of New York at Buffalo
306 Davis Hall
Buffalo, NY, 14260, USA

ABSTRACT

Emerging data that track the dynamics of large populations bring new potential for understanding human decision-making in a complex world and supporting better decision-making through the integration of continued partial observations about dynamics. However, existing models have difficulty with capturing the complex, diverse, evolving, and partially unknown dynamics in social networks, and with inferring system state from isolated observations about a tiny fraction of the individuals in the system. To solve real-world problems with a huge number of agents and system states and complicated agent interactions, we propose a stochastic kinetic model that captures complex decision-making and system dynamics using atomic events that are individually simple but together exhibit complex behaviors. As an example, we show how this model offers significantly better results for city-scale multi-objective driver route planning in significantly less time than models based on deep neural networks or co-evolution.

1 INTRODUCTION

Data that continuously track the dynamics of large populations increase in size rapidly (Blondel et al. 2015), which promotes research into understanding human decision-making in a complex world and supporting better decision-making through information integration. For example, datasets that track vehicles and people are increasingly available for researchers studying sustainable urban development, optimized route plans for drivers, and approaches for relieving the road traffic (Yang et al. 2018).

Researchers have formulated multi-agent decentralized control based on partial observability of the environment as a decentralized partially observable Markov decision process (Dec-POMDP); developed approximate algorithms such as co-evolutive algorithms (Nair et al. 2005), gradient descent for policy search (Peshkin et al. 2000), and Bayesian games (Emery-Montemerlo et al. 2004); and applied these algorithms to applications such as estimation of interactions in a collaborative human-computer environment (Kamar and Grosz 2007), policy search for multi-robot coordination under uncertainty (Amato et al. 2016), and control of the trade-off between accuracy and processing time in video surveillance (Kapoor et al. 2012). Current Dec-POMDP algorithms are promising for many applications, but are rarely applied to solve the multi-agent learning and planning problems in complex systems characterized by a huge number of agents, complex agent interactions and nonlinear dynamics, and changing specifications of optimality over time.

In this paper, we formulate a stochastic kinetic process to model the diverse dynamics of agent interactions and decision-making in a complex network as a sequence of atomic events that individually induce minimal changes to the network but together show diverse behavior. With this formulation, learning and control in a network involve learning to set how fast these interaction events happen in response to the

978-1-7281-3283-9/19/$31.00 ©2019 IEEE

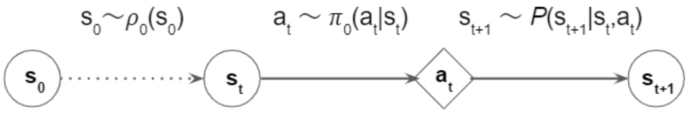


Figure 1: Initial state s_0 is sampled from prior state visitation distribution $\rho_0(s_o)$; action a_t is sampled from the policy π_0 and next state s_{t+1} is chosen based on the transition probability distribution $P(s_{t+1}|s_t,a_t)$.

noisy signals from within and outside the network. Such networked and event-based control often appears in networked social, biological, and engineered systems, which need to solve different optimization problems at different times involving the same sets of subproblems. As a result, these systems prefer modularized design so that they can reorganize modules to quickly solve a new problem. They also need to be robust in that their solutions should not be sensitive to noise from the environment and the network. Various methods such as negative feedback loop and self-validation can help to achieve the robustness goal. As such, the networked design is more adaptive and robust in comparison to a monolithic design that models policy and value function as functions of the system state or past observations.

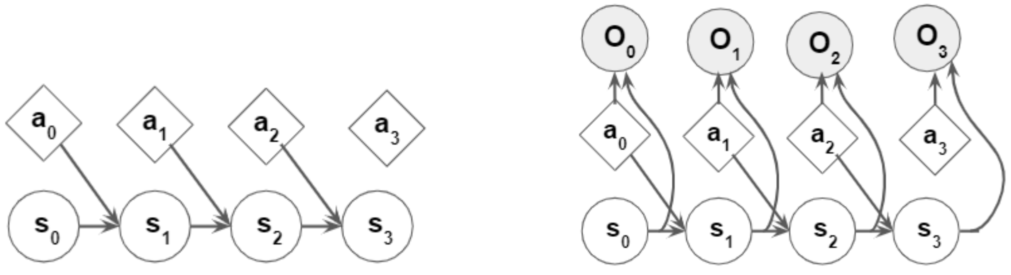
To solve the learning and control problems of a stochastic kinetic model, we reduce these problems to parameter-learning and inference problems in a mixture of dynamic Bayesian networks (Vlassis and Toussaint 2009). With this reduction we can bring in many existing parameter-learning and statistical inference techniques for optimizing the interactions of a networked system based on partial observations about the complex environment (Xu et al. 2016; Yang and Dong 2018), and integrate signal processing and decision-making into a holistic framework. Specifically, we develop a particle filter algorithm to model how the networked system continually tracks the current state of itself and the environment using noisy observation streams, and to learn how to make near-optimal plans by adjusting the rates at which interaction events happen. In this sense, our learning algorithm works through policy search that maximizes the expected log-likelihood over agent policies in a mixture of dynamic Bayesian networks. This is different from the policy gradient method in that the latter uses a stochastic gradient with function approximation, which puts extra constraints on the “compatible” features to represent policy and ensure unbiased gradient estimation (Sutton et al. 2000).

The rest of the paper is structured as follows. We begin with discussing the preliminary material and methodology in Section 2. In Section 3, we give the results of our case study. We conclude with a discussion on open problems in Section 4.

2 BACKGROUND

2.1 Markov Decision Process

In this section, we will introduce the notation we will use for the standard optimal control or reinforcement learning formulation. We consider infinite-horizon partially observable Markov decision process (POMDP), defined by the tuple $\langle S,A,O,P,r,\rho_0,\gamma\rangle$, where $s \in S$ denotes states, describing the possible configurations of all agents; $a \in A$ denotes actions, which can be discrete or continuous; $P : S \times A \times S \to \mathbb{R}$ is the states transition probability distribution, where states evolve according to the stochastic dynamics $p(s_{t+1}|s_t,a_t)$, which are in general unknown; O is a set of observations for each agents; $r : S \to \mathbb{R}$ is the reward function; $\rho_0 : S \to [0,1]$ is the distribution of the initial state s_0; $\gamma \in [0,1]$ is a discount factor (Figure 1). Graphical models for control problems are represented in Figure 2. For our work we make assumptions about the stochastic action choices, in which actions may be chosen with any probabilities to facilitate more robust prediction and planning. To choose an action, each agent uses a stochastic policy $\pi_\theta : O \times A \to [0,1]$, which produces the next state according to the state transition probability. Each agent obtains rewards as a function of the state and agents action $r : S \times A \to \mathbb{R}$, and receives a private observation correlated with the state $\mathbf{o} : S \to O$.



(a)

(b)

Figure 2: The graphical models for the control problem. (a) Graphical model representation and (b) graphical model representation with observation.

Solving a MDP means finding a policy π^* that maximizes the expected long-term reward $R = \sum_{t=0}^{T} \gamma^t r^t$, where T is the time horizon (Sutton and Barto 2018).

3 METHODOLOGY

There are several key challenges in solving a complex real-world planning problem: the number of agents and system states is huge, the agent interactions are complicated, and both the environment and the specification of optimality are constantly changing. Existing planning algorithms generally cannot cope with these challenges. Our solution is to introduce the social stochastic kinetic model. This model captures the complex and diverse social interaction dynamics with a set of atomic events (social interactions) that change the states of individuals only minimally according to simple rules but in sequence can generate complex and diverse dynamics. Optimal control in a social stochastic kinetic process is implemented by adjusting the speeds of the atomic events in response to signals from within and outside this process. We can then develop machine learning algorithms to learn to set optimal event rates from partial or indirect observations about the individuals in the system and the environment.

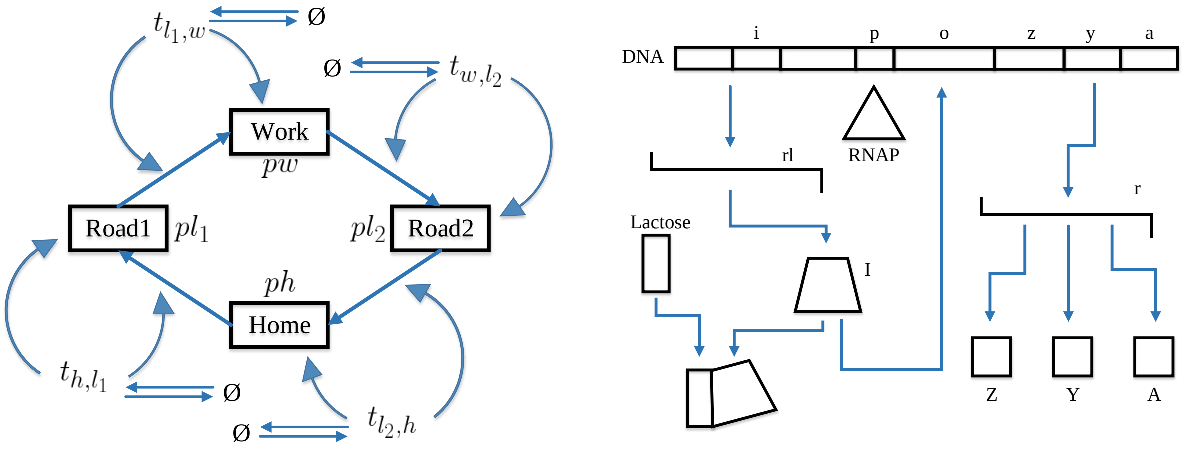
3.1 Stochastic Kinetic Model

A *stochastic kinetic model* (Gillespie 2007; Wilkinson 2011) is a biochemist's way of describing the dynamics of a biological network of M species and V mutually independent interaction events, where the stochastic effects are particularly prevalent – such as a transcription network or signal transduction network. An event (chemical reaction) v is specified by a production like the following:

$$\alpha_v^{(1)}\mathbf{X}^{(1)} + \cdots + \alpha_v^{(M)}\mathbf{X}^{(M)} \overset{c_v}{\to} \beta_v^{(1)}\mathbf{X}^{(1)} + \cdots + \beta_v^{(M)}\mathbf{X}^{(M)}, \tag{1}$$

where $\mathbf{X}^{(1)}, \dots, \mathbf{X}^{(M)}$ symbolize the individuals (molecules) of the M species in the network. The production is interpreted as having rate c_v (probability for event to happen per unit of time), at which $\alpha_v^{(1)}$ individuals of species 1, $\alpha_v^{(2)}$ individuals of species 2 interact according to event v, resulting them being removed from the system, and $\beta_v^{(1)}$ individuals of species 1, $\beta_v^{(2)}$ individuals of species 2 ... being introduced into the system. As such, event v changes the populations by $\Delta_v = (\beta_v^{((1)} - \alpha_v^{(1)}, \dots, \beta_v^{(M)} - \alpha_v^{(M)})$.

Let $x_t = (x_t[1], \dots, x_t[M])$ be the populations of the M species in the system at a discrete time t, τ be an infinitesimal time step, and $\emptyset$ be an auxiliary event that does not change the populations. The time-discretized stochastic kinetic process initially in state x_0 at time $t = 0$ can be simulated through the Gillespie algorithm (Gillespie 2007) by iteratively (i) sampling the event $v \in \{\emptyset, 1, \dots, V\}$ according to categorical distribution $v \sim (1 - \tau h_0, \tau h_1, \dots, \tau h_V)$, where $h_v(x_t, c_v)$ is the rate of event v and $h_0(x, c) = \sum_{v=1}^{V} h_v(x_t, c_v)$ is the rate of all events, and (ii) updating the populations $x \leftarrow x + \Delta_v$ accordingly, until the termination



(a)

(b)

Figure 3: Complex interaction dynamics and decision-making in a social network. (a) Control in road network and (b) control in lac-operon.

condition is satisfied. In this algorithm, the event rate $h_v(x_t, c_v)$ is the constant rate c_v multiplying a total of $\prod_{m=1}^{M}(x_t^{(m)})^{\alpha_v^{(m)}}$ different ways for individuals to interact in the system, assuming homogeneous populations. Exponential distribution is the maximum entropy distribution given the rate constant, and consequently is favored by nature. The stochastic kinetic model thus assigns a probability (2) to a sample path induced by a sequence of events $v_1, \ldots, v_T$, where δ is an indicator function.

$$P(v_{1:n}, x_{0:n}) = p(x_0) \prod_t p(v_{t+1}|x_t) \cdot \delta(x_{t+1}, x_t + \Delta_{v_{t+1}}), \tag{2}$$

where

$$p(v_{t+1}|x_t) = \begin{cases} 1 - \tau \cdot h_0(x_t), & v_{t+1} = \emptyset \\ \tau \cdot h_k(x_t), & v_{t+1} = k, \end{cases} \tag{3}$$

$$h_v(x, c_v) = c_v g_v(x) \text{ for } v = 1, \ldots, V, \tag{4}$$

$$h_0(x, c) = \sum_{v=1}^{V} h_v(x, c_v). \tag{5}$$

To model how complex networks achieve optimal control by setting the appropriate speeds of the atomic interaction events in response to noisy signals from within and outside the network, we specify that the *observations* $y_t = y_t[1], \ldots, y_t[M]$ on the populations are conducted independently (8), and that the expected *reward* of the network is the total expected reward assigned to the individuals of the different species ($r_t[m]$). Our goal is to learn the event rates $h_v(x, c_v)$ (4) in order to maximize the expected future reward of the network conditioned on past observations $y_{-\infty,\ldots,0}$ according to the probability model $p(v_{1:T}, x_{0:T}, y_{1:T})$ (7).

$$\arg\max_{c_{1:V}} \mathbf{E}_{x_{0:\infty}, v_{0:\infty}|y_{-\infty:0}} \left(\sum_{t=0}^{\infty} \gamma_t \sum_{m=1}^{M} x_t[m] \cdot r_t[m]; c_{1:V} \right), \tag{6}$$

where

$$p(v_{1:T}, x_{0:T}, y_{1:T}) = p(v_{1:T}, x_{0:T}) \prod_{t=1}^{T} p(y_t|x_t), \tag{7}$$

$$p(y_t|x_t) = \prod_{m=1}^{M} p(y_t[m]|x_t[m]). \tag{8}$$

We use the gene regulation of the lac operon (Jacob and Monod 1961) (the first well-understood genetic regulatory network, shown in Figure 3b) to illustrate how a biological network optimizes its *fitness* *function* by selecting the right network dynamics according to the environment. The genes in the lac operon code for enzymes for bacteria to ”burn” lactose for energy when glucose – the preferred ”fuel” – is in short supply. Under normal operation conditions, transcription of the lac operon is turned off by an inhibitor protein bound to the operon to conserve energy ($i \to r_I$, $r_I \to r_I + I$ and $I + o \to I \cdot o$, where i represents the gene for the inhibitor protein, r_I the associated mRNA, I the inhibitor protein and o the lac operon). However, in the presence of lactose the inhibitor preferentially binds to lactose and releases the lac operon to produce enzymes to burn lactose ($\text{lactose} + I \to \text{lactose} \cdot I$, $o \to o + r$, $r \to r + X + Y + Z$ and $Z + \text{lactose} \to Z + \text{energy}$, where r denoted the mRNA transcript from the lac operon, and X, Y, Z the three lac proteins coded by r). For growing bacteria, it needs to balance the energy used between duplicating and producing proteins X, Y, Z (where Z will produce more energy when lactose is abundant). The fitness function corresponds to the cell growth rate, and optimal network dynamics that maximize bacteria growth rate are reached rapidly and precisely by natural selection as demonstrated in controlled experiments (Dekel and Alon 2005). Here, network dynamics include the right amount of protein molecules to produce, whether or not to regulate, and the type of regulation (Alon 2006).

The central idea of this research is that complex dynamics and decision-making in a social network can similarly be expressed as a sequence of atomic social interactions (Yang et al. 2019; Dong et al. 2019). For example, we can use $pl_1 + t_{l_1,l_2} \to pl_2$ to express that a person p moves from location (building or road) l_1 to a neighboring location l_2 according to a traffic information token t_{l_1,l_2}. The traffic information tokens are always expressed, $\emptyset \to t_{l_1,l_2}$ and $t_{l_1,l_2} \to \emptyset$, with their generation rates and degradation rates depending on time (Figure 3a). They can be influenced by traffic on the corresponding roads – for example, a crowded location will have traffic tokens pointing to this location removed from the system and as a result the location will be visited less: $t_{l_1,l_2} + pl_2 \to pl_2$. They can form more elaborate interactions to filter out short pulses in traffic conditions through a feedforward loop, to implement a sequence of controls through a fanout structure, or to shorten response time and achieve robust control through negative self-regulation (Alon 2006).

Executing a policy in this road traffic network involves estimating the driver populations $x_t[l]$ at all locations l at time t and the traffic token populations $x_t[t_{l_1,l_2}]$ for all location transitions from l_1 to l_2 at time t from noisy past observations $y_{-\infty,t}[l]$ about the driver populations at some locations l and some times t. The optimal policy maximizes the total expected future reward over all drivers (Horni et al. 2016) for all observation histories. In comparison with how actions control state transition probabilities in a standard Markov decision process, a stochastic kinetic process controls the state transitions of the drivers through the traffic token populations, whose stochastic interaction with the drivers determine the drivers' state transitions. In comparison with how a policy prescribes the next action according to the current state in a standard Markov decision process, a stochastic kinetic process sets the traffic token populations through their interactions with the driver populations.

3.2 Optimal Control with Particle Filter

In this subsection, we reduce learning and planning of a partially observable Markov decision process to parameter learning and latent state inference of a mixture of dynamic Bayesian networks (Vlassis and Toussaint 2009; Toussaint 2009), and develop particle-based algorithms to learn near-optimal policy.

The equivalence between the expected future reward of a Markov decision process and the probability of receiving a reward in a mixture of dynamic Bayesian networks is shown in the following derivation:

$$
\begin{aligned}
J =& \mathbf{E}\left(\sum_{t=0}^{\infty} \gamma_t \sum_{m=1}^{M} x_t[m] \cdot r_t[m] \right) && (9) \\
& \text{choose } p(t), p(T) \text{ such that } \gamma_t \propto \sum_{T=0}^{\infty} p(T) p(t) \delta(t \leq T) \\
\propto & \sum_{T=0}^{\infty} p(T) \sum_{t=0}^{T} p(t) \sum_{m=1}^{M} \mathbf{E}(x_t[m]) r_t[m] && (10) \\
\propto & \sum_{T=0}^{\infty} p(T) \sum_{t=0}^{T} p(t) \sum_{m=1}^{M} \mathbf{E}(x_t[m]) p(R_t[m] = 1). && (11)
\end{aligned}
$$

Equation (11) identifies the value function J up to a scaling factor with the probability for any agent to receive the binary reward over the tuples $(T, t, m, x_{1:t}[m], R_t[m])$: $J \propto p_{T,t,m,x_{1:t},R_t[m]}(R_t[m] = 1)$. In Equation (10), we can select $p(T) = (1-\delta)\delta^t$ and $p(t) = (1-\frac{\gamma}{\delta})(\frac{\gamma}{\delta})^t$ for discounted future rewards with $\gamma_t = \gamma^t$, and $p(T) = \delta_{H,T}$ and $p(t) = 1/(H+1)$ for finite-horizon future rewards with $\gamma_t = \delta_{t \leq H}$, where δ is an indicator function. In Equation (11), we redefine the reward as a Bernoulli trial, with its probability of success being proportional to the expected reward, $p(R_t[m] = 1) \propto r_t[m]$.

In a stochastic kinetic process, the latent state is comprised of the actor populations and the control populations. We use a particle filter to model how a stochastic kinetic process executes a policy by iteratively proposing actor and control populations (i.e., particles) from the learned dynamics, and then selecting the most likely populations according to new observations about the actor populations. Specifically, let x_t^k for $k = 1, \dots, K$ be a collection of particle positions and v_t^k the corresponding events from particle mutation, and $i_t^k \in \{1, \dots, K\}$ be the collection of particle indexes from particle selection. To make inferences about the latent state x_t of a stochastic process starting at state x_0 from observations $y_{1:t}$, we initialize particle positions and indexes as $x_0^1, \dots, x_0^K = x_0$ and $i_0^1 = 1, \dots, i_0^K = K$, iteratively sample the next event v_t^k according to how likely it is that different events will occur conditioned on system state $x_{t-1}^{i_{t-1}^k}$ for $k = 1, \dots, K$ (12), and then update $x_t^k = x_{t-1}^k + \Delta_{v_t^k}$ accordingly (13) and resample these events per their likelihoods with regard to the observation y_t for $t = 1, \dots, T$ (14). The expected reward at time t is $\frac{1}{K} \sum_{k=1}^{K} \sum_{m=1}^{M} x_t^{i_t^k}[m] p(R_t[m] = 1)$. Here, we specify that the control populations are not observable and no reward is given to individuals in the control populations, $p(y_t[m] | x_t[m]) = 1$ and $r_t[m] = p(R_t[m]) = 0$ for all control populations m.

$$
\begin{aligned}
& v_t^k | x_{t-1}^{i_{t-1}^k} \sim \text{Categorical}(1 - \tau h_0, \tau h_1, \dots, \tau h_V)(x_{t-1}^{i_{t-1}^k}), && (12) \\
& x_t^k = x_{t-1}^{i_{t-1}^k} + \Delta_{v_t^k}, && (13) \\
& i_t^k | (x_t^{1:N}, y_t) \sim \text{Categorical}(p(y_t | x_t^1), \dots, p(y_t | x_t^N)). && (14)
\end{aligned}
$$

To determine a particle trajectory from the posterior distribution of a stochastic kinetic process with respect to observations, we trace back the events that lead to the particles $x_T^{i_T^k}$ for $k = 1, \dots, N$:

$$
x_0, v_1^{j_1^k}, x_1^{j_1^k}, \dots, v_T^{j_T^k}, x_T^{j_T^k}, \text{where } j_T^k = i_T^k, j_{T-1}^k = i_{T-1}^{j_T^k}, \dots, j_1^k = i_1^{j_2^k}. \qquad (15)
$$

Policy in a stochastic kinetic model is parameterized by event rate constants, and policy search involves identifying optimal rate constants from a training data set of historical observations of actor populations. To this end, we sample particle trajectories from the historical observations according to Equations (12, 13, 14, and 17), and maximize the expected log likelihood for a mixture component to receive the binary reward, where the expectation is taken over the posterior probability of the mixture components conditioned on that a reward is received and is approximated by the particle trajectories. Intuitively, the maximum likelihood estimation of the rate constants over a single mixture component is $c_v = \sum_t \delta(v, v_t) \big/ \sum_t \tau g(x_{t-1})$ because the expected number of events to happen over the trajectory $\sum_t \tau c_v g(x_{t-1})$ should match the number of events that happened $\delta(v, v_t)$. The maximum expected log likelihood estimation of the rate constants (18) replaces all statistics in the maximum log likelihood estimation with expectations.

$$c_v = \frac{\sum\limits_{k,t'} \delta(v, v_{t'}^k) \sum\limits_{t=t'}^{T} p(t) \sum\limits_{m=1}^{M} \mathbf{E}(x_t^k[m]) p(R_t[m]=1)}{\sum\limits_{k,t'} \tau g(x_{t'-1}^k) \sum\limits_{t=t'}^{T} p(t) \sum\limits_{m=1}^{M} \mathbf{E}(x_t^k[m]) p(R_t[m]=1)}. \tag{16}$$

To summarize, we have developed Algorithm 1 to execute a given policy and Algorithm 2 to update the policy in light of observation history training data. The overall architecture is also shown in Figure 4. The algorithm is done in two parts. During the Inference we estimate the state density and reward associated with each particle. During the Search step we update policy parameters according to the Inference results. The algorithm iterates until converge. When the complex system has an extremely high dimension, variational inference algorithms (Yang et al. 2019) should be considered to avoid particle degeneracy issues. However, as our experiment shows, the particle-based algorithm works well with social systems under general considerations.

```
Algorithm 1: Inference with Stochastic Kinetic Model
  Input  : SKM model defined by Equation (7). Observation stream $\{y_t | t \geq 0\}$
  Output: particles at time $t$: $x_t^{i_t^1}, \dots, x_t^{i_t^K}$.
  Initialize $x_0^{i_0^1}, \dots, x_0^{i_0^K}$
  for $t = 1, 2, \dots$ do
      execute Eqs. 12, 13 and 14.
  end
```

```
Algorithm 2: Policy search with Stochastic Kinetic Model
  Input  : SKM model defined by Equation (7). Observation history training data $\{y_t | t = 0 : T\}$
  Output: optimized event rate constants $c_1, \dots, c_V$.
  Iterate through E-step and M-step until convergence.
    • E-step: sample particle trajectory
        $x_0, v_1^{j_1^k}, x_1^{j_1^k}, \dots, v_T^{j_T^k}, x_T^{j_T^k}$, where $j_T^k = i_T^k, j_{T-1}^k = i_{T-1}^{j_T^k}, \dots, j_1^k = i_1^{j_2^k}$.   (17)
    • M-step: update $c_1, \dots, c_V$ of current SKM.
        $c_v = \frac{\sum_{k,t'} \delta(v, v_{t'}^k) \sum_{t=t'}^{T} p(t) \sum_{m=1}^{M} \mathbf{E}(x_t^k[m]) p(R_t[m]=1)}{\sum_{k,t'} \tau g(x_{t'-1}^k) \sum_{t=t'}^{T} p(t) \sum_{m=1}^{M} \mathbf{E}(x_t^k[m]) p(R_t[m]=1)}.$   (18)
```

4 CASE STUDY

In this section, we evaluate the performance of a Stochastic Kinetic Model with the problem of optimizing travel plans for all drivers in a city-scale transportation network by using observations from probe vehicles in limited locations. The driver plans are evaluated using a multi-objective reward function: each individual receives a penalty for every minute spent on the road, a reward when he works at the expected working time, and a penalty if he arrives late or leaves early (Horni et al. 2016).

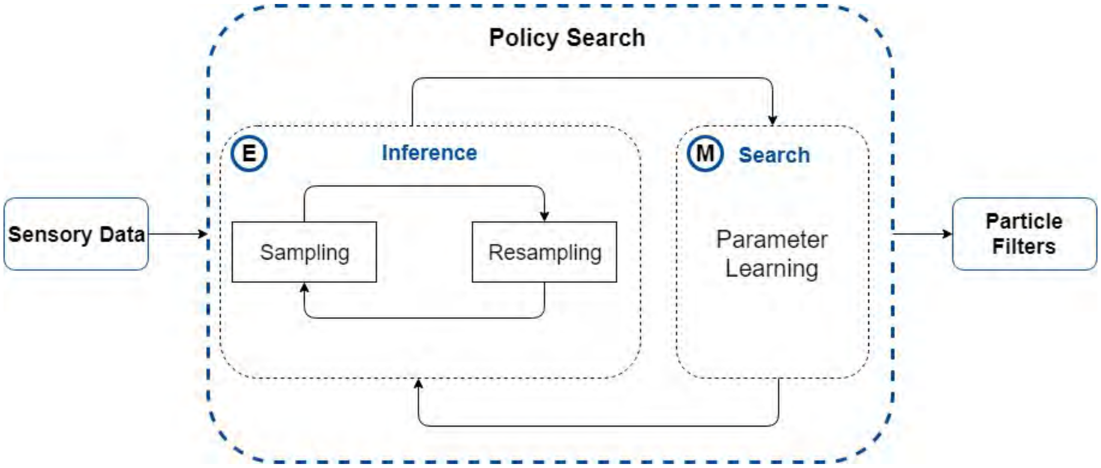


Figure 4: Overview of our policy search algorithm.

4.1 Stochastic Kinetic Model of Traffic Dynamics

In this application, each agent represents a vehicle, whose observation is probe vehicles on limited locations. The goal is to find the optimal travel policy of the system. We model road traffic dynamics through one type of event (19)

$$pl_1 + t_{l1,l2} \to pl_2, \tag{19}$$

where a vehicle p moves from location l_1 to location l_2 with information token $t_{l1,l2}$. We use three types of events to manipulate the population of information token: $\emptyset \to t_{l1,l2}$ and $t_{l1,l2} \to \emptyset$, the information token can be generated and removed with rate according to time; $t_{l1,l2} + pl_2 \to pl_2$, tokens pointing to a crowded location will be removed and hence less vehicles moving to such location.

The optimal control in the traffic network is achieved through adjusting the vehicle moving event rates in response to the observations. Each vehicle moving event is associated with a control species, which population is proportional to the event rates. We use three types of events to modify the population of control species: $\emptyset \xrightarrow{v_j} c_j$, a new individual of control species c_j be generated with event rate v_j; $c_j \xrightarrow{v_k} \emptyset$, an individual of control species c_j disappears with event rate v_k; $c_j + s_j \xrightarrow{v_l} s_j + c_k$, with probability v_l an individual of control species c_j changes to another c_k on occurrence of vehicles on the road where the vehicle moving event of control species c_j leads to.

Each agent is associated with a set of control species, which regulate the agent moving event rates. The population of the control species is proportional to the event rates. We use three types of events to modify the population control species: $\emptyset \to c_j$, a new control species be generated; $c_j \to \emptyset$, a control species disappears; $c_j + s_j \to s_j + c_k$, a control species c_j changes to another c_k if there are too many vehicles on this location, hence reducing the event rate.

4.2 Datasets Descriptions and Evaluation Metrics

The performance of the SKM against other algorithms was evaluated by using two datasets. The SynthTown dataset is comprised of a synthesized network that includes one home location, one work location, and 23 single-direction road links, and the trips of $2,000$ synthesized inhabitants going to work in the morning and returning home in the evening. We use this dataset to compare different algorithms in detail. The Berlin data set is comprised of a network of $24,335$ single-direction road links and the trips of $9,178$ synthesized vehicles representing the travel behavior of one million vehicles (Ziemke et al. 2015). We use this dataset to gauge the scalability of different algorithms in working with more complex dynamics.

Four metrics were used to evaluate different planning algorithms. *Average trip time* is measured in minutes of all vehicles driving from home to work, and a lower average trip time means better traffic. *On-time arriving ratio* measures the percentage of people arriving to work on time. *Expected reward* is measured per vehicle per hour, and higher expected rewards demonstrate better individual plans and more efficient transportation network. *Training epochs* measure the number of training epochs needed for the algorithm to converge. These metrics show both the behaviors of the algorithms and the perceptions of the agents in the complex system.

4.3 Benchmark Algorithms

Our SKM is compared with a baseline algorithm, a co-evolutionary algorithm, and a neural network policy gradient algorithm. The baseline algorithm (Baseline) optimizes agents' expected future rewards without considering the current traffic situation or the plans of other agents. The co-evolutionary algorithm (CoEAs) is the state-of-the-art algorithm for generating the equilibrium of daily activities and trips in transportation theory (Horni et al. 2016). In CoEAs, agents independently explore and exploit their plans through a genetic operator, jointly execute and evaluate their plans in a simulator, and repeat this process until equilibrium is reached (Popovici et al. 2012). The neural network policy gradient algorithm (NNPG) is an approximate planning algorithm that maximizes $\sum_k V_k \log p(a_k|x_k;w)$ over synaptic weights w, where $p(a_k|x_k) = \text{NNPG}(x_k;w)$ is the neural network output and (x_k,a_k,V_k) is a tuple of input, action, and value (Silver et al. 2016). The benchmark neural network has four layers. The input layer receives the current time and the minute-by-minute probe vehicle counts in selected locations at specific times within the past hour, and feeds these values into the three hidden layers.

4.4 Results

Comparing benchmark algorithms on SynthTown dataset (Figure 5a), the baseline algorithm distributed most of the traffic at road links 6 and 15, and the rest at road links 2, 3, 11, and 12 (Figure 5b). On the other hand, our algorithm distributed the traffic almost evenly among road links 2 - 19 (Figure 5c). Thus, all road links were used by commuters to reach work.

Figure 6 is a heat map representation of algorithm performances in the SynthTown scenario. The x-axis indicates the hour of a day, the y-axis shows the road links. The brighter yellow indicates that there is a higher level of traffic, and, when red, the opposite. The white vertical line shows when people are suppose to arrive at work, and the black line shows when all vehicles actually arrive at work. For all figures, people are supposed to be at work at 9am, but the actual arrival time differs. With the baseline algorithm (Figure 6a), people are jammed at road links 6 and 15, and arrive at work at 4pm. With neural network policy gradient, people can arrive at work at 10am (Figure 6b). Our algorithm resulted in the most even traffic distribution, and navigate people to work on time at 9am 6c.

Table 1 compares the average trip time, on-time arriving ratio, and average unit reward statistics of the four models with the SynthTown and Berlin datasets. The Berlin dataset is too large for the NNPG model to run, which indicates the better scalability of SKM and CoEAs. This comparison indicates that SKM has the lowest average trip time, highest on-time arrival ratio, highest expected reward, and smallest training epochs in all datasets. First, SKM outperforms NNPG because it has better scalability. Second, SKM outperforms CoEAs because our model factors complex and diverse social interaction dynamics into a set of atomic events, which makes it more robust and resistant to noise. Third, MDEDP converges fastest because it reduces the procedure of searching for the optimal policy to the optimization of event rates in a sequence of atomic events.

5 CONCLUSIONS

In this paper, we formulated a stochastic kinetic process model to specify the complex interactions and decision-making in a real-world social network using a set of atomic social interaction events. Based on

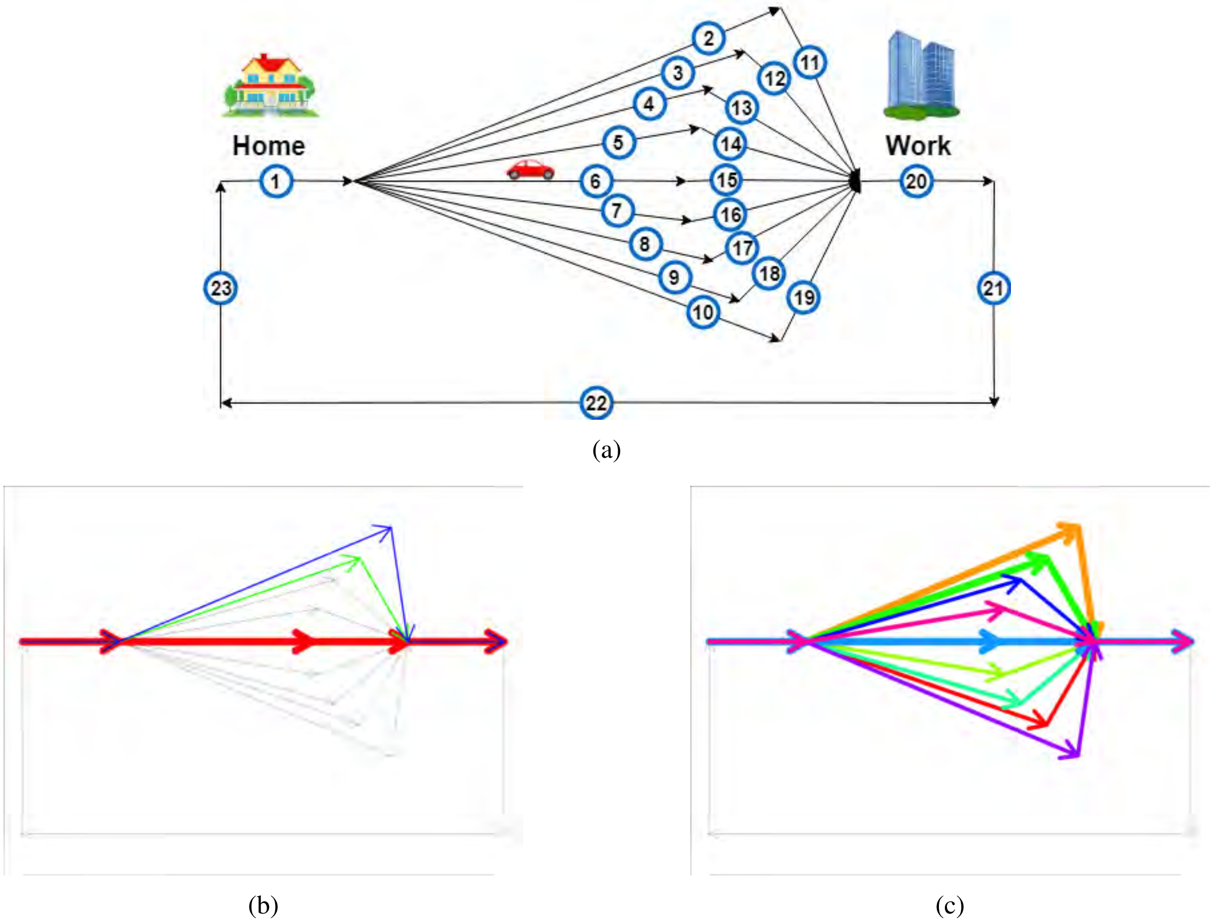


Figure 5: (a) SynthTown road links. There are 23 road links in total, where segment 1 indicates ”Home” and segment 20 indicates ”Work”. Traffic distribution among road links after applying algorithms, where (b) is the baseline and (c) is our algorithm.

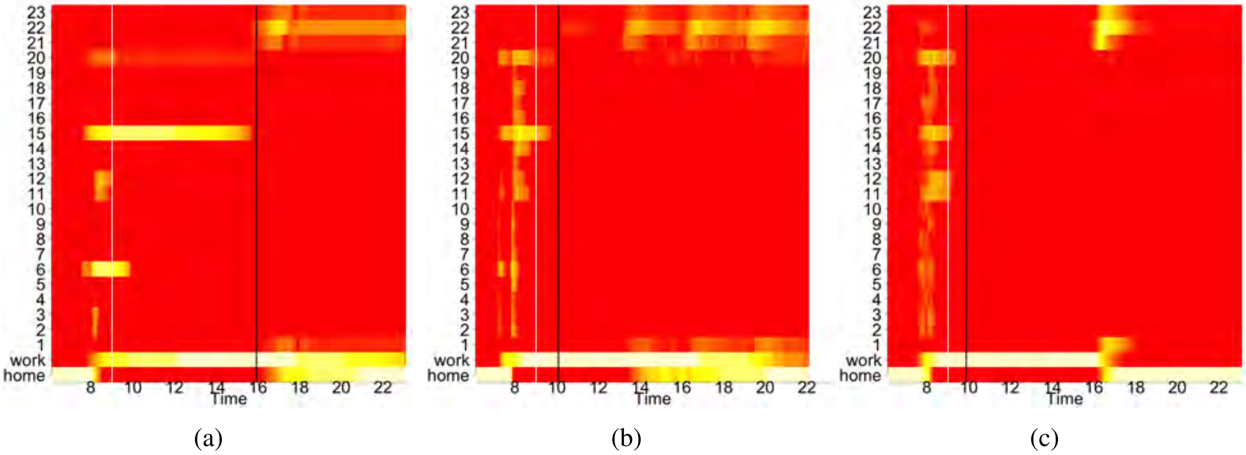


Figure 6: Heatmap representations of different approaches to solve the optimization problem. (a) baseline, (b) NNPG and (c) our algorithm - SKM.

Table 1: Comparison results.

Dataset	Models	Average trip time	On-time arriving ratio	Expected reward	Training epochs
SynthTown	Baseline	161.46	0.29	-252.78	
	SKM	**31.49**	**0.89**	**2.93**	**20**
	CoEAs	55.47	0.85	-0.05	200
	NNPG	128.36	0.88	-85.33	100
Berlin	Baseline	42.72	0.44	-723.33	
	SKM	**38.38**	**0.86**	**-4.83**	**20**
	CoEAs	40.27	0.68	-540.00	200

the equivalence between the expected future reward of a partially observable Markov decision process and the probability of receiving a binary reward in a mixture of dynamic Bayesian networks, we reduced the problem of controlling a POMDP to inferring the probability distributions of latent controlling variables, and the problem of learning the optimal policy to learning the control parameters. The networked discrete event control in the stochastic kinetic model offers significantly better policies in significantly less time than models that treat control as a monolithic complex function to be learned.

REFERENCES

Alon, U. 2006. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Boca Raton, Florida: CRC Press.

Amato, C., G. Konidaris, A. Anders, G. Cruz, J. P. How, and L. P. Kaelbling. 2016. "Policy Search for Multi-robot Coordination under Uncertainty". *The International Journal of Robotics Research* 35(14):1760–1778.

Blondel, V. D., A. Decuyper, and G. Krings. 2015. "A Survey of Results on Mobile Phone Datasets Analysis". *arXiv preprint arXiv:1502.03406*.

Dekel, E. and U. Alon. 2005. "Optimality and Evolutionary Tuning of the Expression Level of a Protein". *Nature* 436(7050):588–592.

Dong, W., T. Guan, B. Lepri, and C. Qiao. 2019. "PocketCare: Tracking the Flu with Mobile Phones Using Partial Observations of Proximity and Symptoms". *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3(2):41:1–41:23.

Emery-Montemerlo, R., G. Gordon, J. Schneider, and S. Thrun. 2004. "Approximate Solutions for Partially Observable Stochastic Games with Common Payoffs". In *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multiagent Systems*,-Volume 1. July 19th–23rd, New York, New York, 136–143.

Gillespie, D. T. 2007. "Stochastic Simulation of Chemical Kinetics". *Annual Review of Physical Chemistry* 58:35–55.

Horni, A., K. Nagel, and K. W. Axhausen. 2016. *The Multi-Agent Transport Simulation MATSim*. London: Ubiquity Press.

Jacob, F. and J. Monod. 1961. "On the Regulation of Gene Activity". In *Cold Spring Harbor Symposia on Quantitative Biology*, Volume 26, 193–211. Cold Spring Harbor, New York: Cold Spring Harbor Laboratory Press.

Kamar, E. and B. Grosz. 2007. "Applying MDP Approaches for Estimating Outcome of Interaction in Collaborative Human-Computer Settings". In *Proceedings of the Multi-Agent Sequential Decision Making in Uncertain Domains (MSDM) Workshop*. May 14th-18th, Honolulu, Hawaii.

Kapoor, K., C. Amato, N. Srivastava, and P. Schrater. 2012. "Using POMDPs to Control an Accuracy-processing Time Trade-off in Video Surveillance". In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*. July 22nd-26th, Toronto, Canada, 2293–2298.

Nair, R., P. Varakantham, M. Tambe, and M. Yokoo. 2005. "Networked Distributed POMDPs: A Synthesis of Distributed Constraint Optimization and POMDPs". In *AAAI*, Volume 5, 133–139.

Peshkin, L., K.-E. Kim, N. Meuleau, and L. P. Kaelbling. 2000. "Learning to Cooperate via Policy Search". In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*. June 30th–July 3rd, Stanford, CA, 489–496.

Popovici, E., A. Bucci, R. P. Wiegand, and E. D. De Jong. 2012. "Coevolutionary Principles". In *Handbook of Natural Computing*, 987–1033. Berlin, Heidelberg: Springer.

Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis. 2016. "Mastering the Game of Go with Deep Neural Networks and Tree Search". *Nature* 529(7587):484–489.

Sutton, R. S., and A. G. Barto. 2018. *Reinforcement Learning: An Introduction*. Cambridge, Massachusetts: MIT Press.

Sutton, R. S., D. A. McAllester, S. P. Singh, and Y. Mansour. 2000. “Policy Gradient Methods for Reinforcement Learning with Function Approximation”. In *Proceedings of the 12th Advances in Neural Information Processing Systems*. November 29th-December 4th, Denver, Colorado, 1057–1063.

Toussaint, M. 2009. “Probabilistic Inference as a Model of Planned Behavior”. *Künstliche Intelligenz* 3(9):23–29.

Vlassis, N. and M. Toussaint. 2009. “Model-free Reinforcement Learning as Mixture Learning”. In *Proceedings of the 26th Annual International Conference on Machine Learning*. June 14th–18th, Montreal, Canada, 1081–1088.

Wilkinson, D. J. 2011. *Stochastic Modelling for Systems Biology*. Boca Raton, Florida: Chapman & Hall/CRC Press.

Xu, Z., W. Dong, and S. N. Srihari. 2016. “Using Social Dynamics to Make Individual Predictions: Variational Inference with a Stochastic Kinetic Model”. In *Advances in Neural Information Processing Systems*, edited by D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, and R. Garnett, December 5th–10th, Barcelona, Spain, 2783–2791.

Yang, F., and W. Dong. 2018. “Integrating Simulation and Signal Processing in Tracking Complex Social Systems”. *Computational and Mathematical Organization Theory* 1:1–22.

Yang, F., B. Liu, and W. Dong. 2019. “Optimal Control of Complex Systems through Variational Inference with a Discrete Event Decision Process”. In *Proceedings of the 18th International Conference on Autonomous Agents & Multiagent Systems*. May 13th–17th, Montreal, Canada, 296–304.

Yang, F., A. Vereshchaka, and W. Dong. 2018. “Predicting and Optimizing City-Scale Road Traffic Dynamics Using Trajectories of Individual Vehicles”. In *Proceedings of the 2018 IEEE International Conference on Big Data*. December 10th–13th, Seattle, Washington, 173–180.

Ziemke, D., K. Nagel, and C. Bhat. 2015. “Integrating CEMDAP and MATSim to Increase the Transferability of Transport Demand Models”. *Transportation Research Record: Journal of the Transportation Research Board* (2493):117–125.

AUTHOR BIOGRAPHIES

FAN YANG is a Ph.D. candidate in the Department of Computer Science and Engineering at the State University of New Your at Buffalo, USA. His research interests includes reinforcement learning, imitation learning, modeling, generative models, and probabilistic graphical models. His is email address is fyang24@buffalo.edu. His website is https://cse.buffalo.edu/∼fyang24/.

ALINA VERESHCHAKA is a Ph.D. candidate in the Department of Computer Science and Engineering at the State University of New Your at Buffalo, USA. Her current research interests include deep reinforcement learning, optimization, and multi-agent modeling in stochastic environments. She has conducted studies in the application areas of optimization, transportation, and healthcare. Her email address is avereshc@buffalo.edu. Her website is https://cse.buffalo.edu/∼avereshc/.

WEN DONG is an Assistant Professor of Computer Science and Engineering with a joint appointment at the Institute of Sustainable Transportation and Logistics at the State University of New York at Buffalo. His research focuses on developing machine learning and signal processing tools to study the dynamics of large social systems in situ. He has a Ph.D. degree from the M.I.T. Media Laboratory. His email address is wendong@buffalo.edu. His website is https://cse.buffalo.edu/∼wendong/.